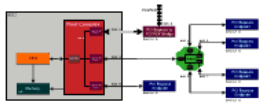
OVERVIEW OF PCI(e) SUBSYSTEM

Author Name
author@email.com

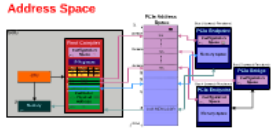
Introduction

Terminology

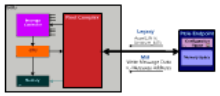
Topology



Address Space

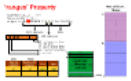


Interrupts

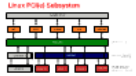


Device Tree

'nouges' Property



Linux PCIe Subsystem



Interrupt Handling

Input

Acknowledgements

References

Happy Hacking!

OVERVIEW OF PCI(e) SUBSYSTEM

KISHON VIJAY ABRAHAM I, VIGNESH R

Introduction

Terminology

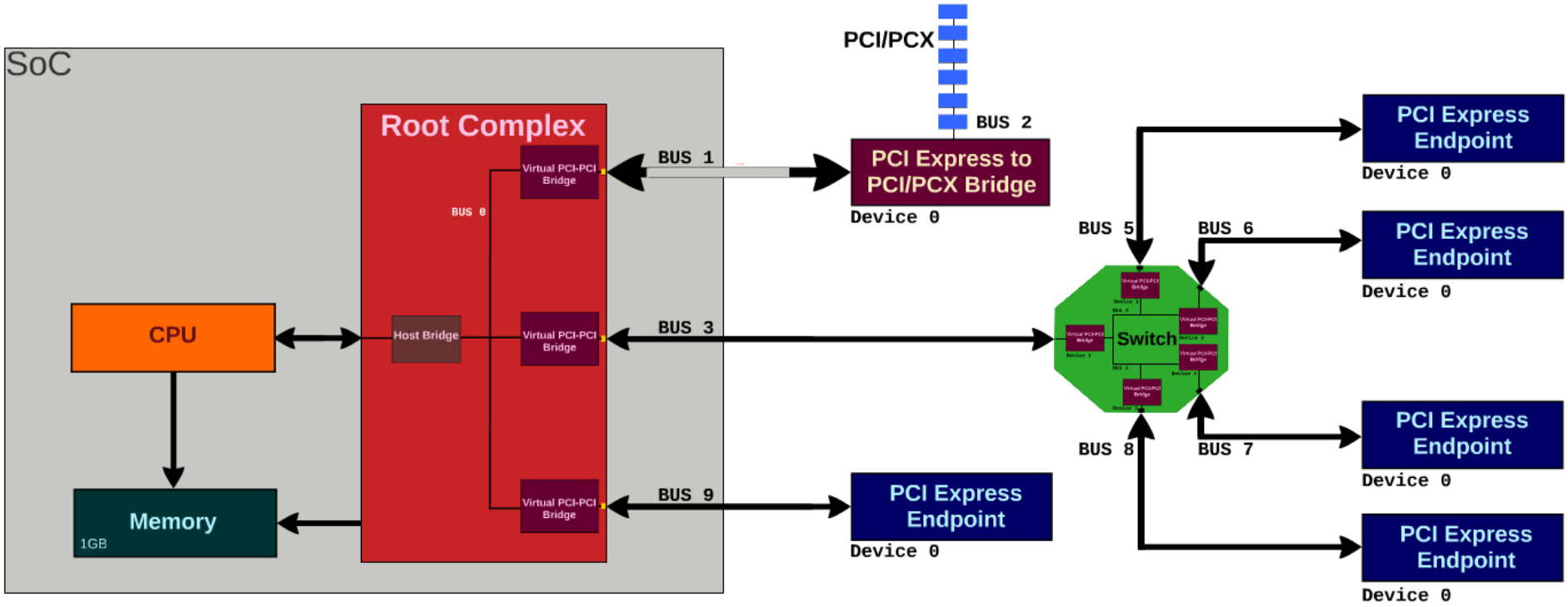
Introduction

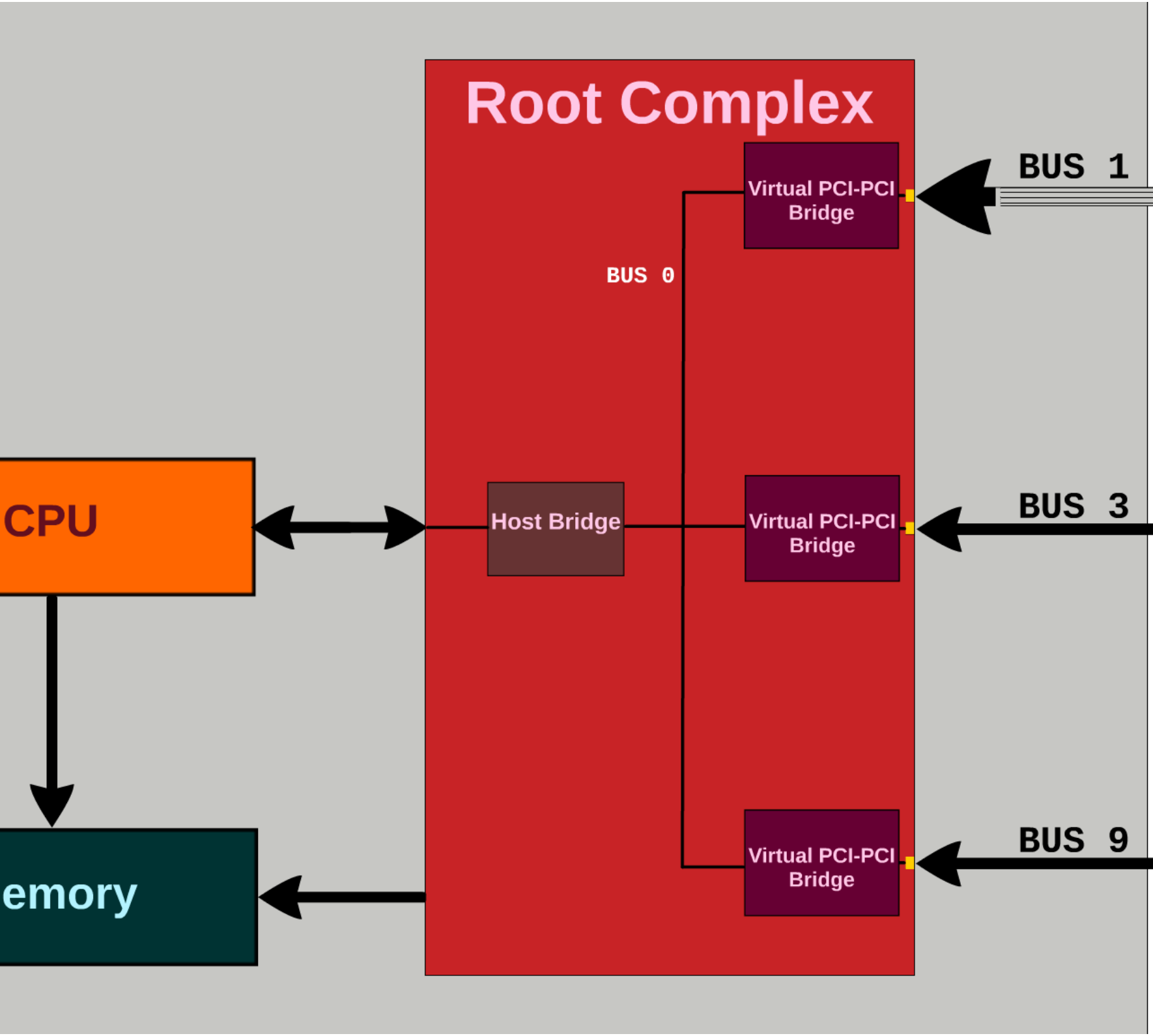
- PCIe Vs PCI
- Usage model and software interface same as PCI
- Serial communication interface like USB, SATA
- Performance: Higher throughput, multi-lane
- Add peripherals to the system (USB, Ethernet, SATA, DCAN etc..)

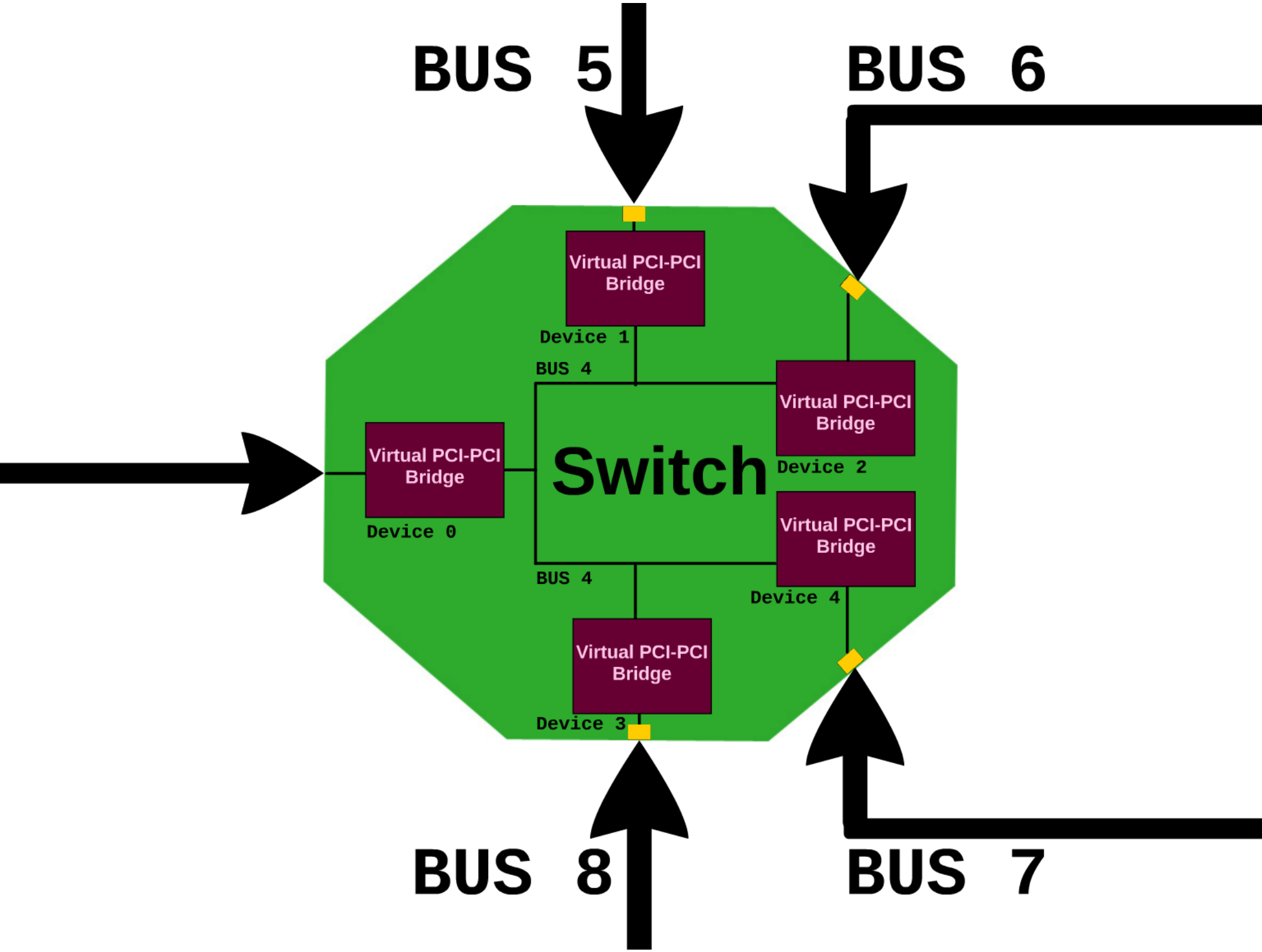
Terminology

- Root Complex - PCIe host
- Endpoint - PCIe device
- cpu_addr - CPU physical address (proc/iomem)
- pci_addr - PCI bus address
- Switches - allow more devices to be connected
- Enumeration - discovering devices

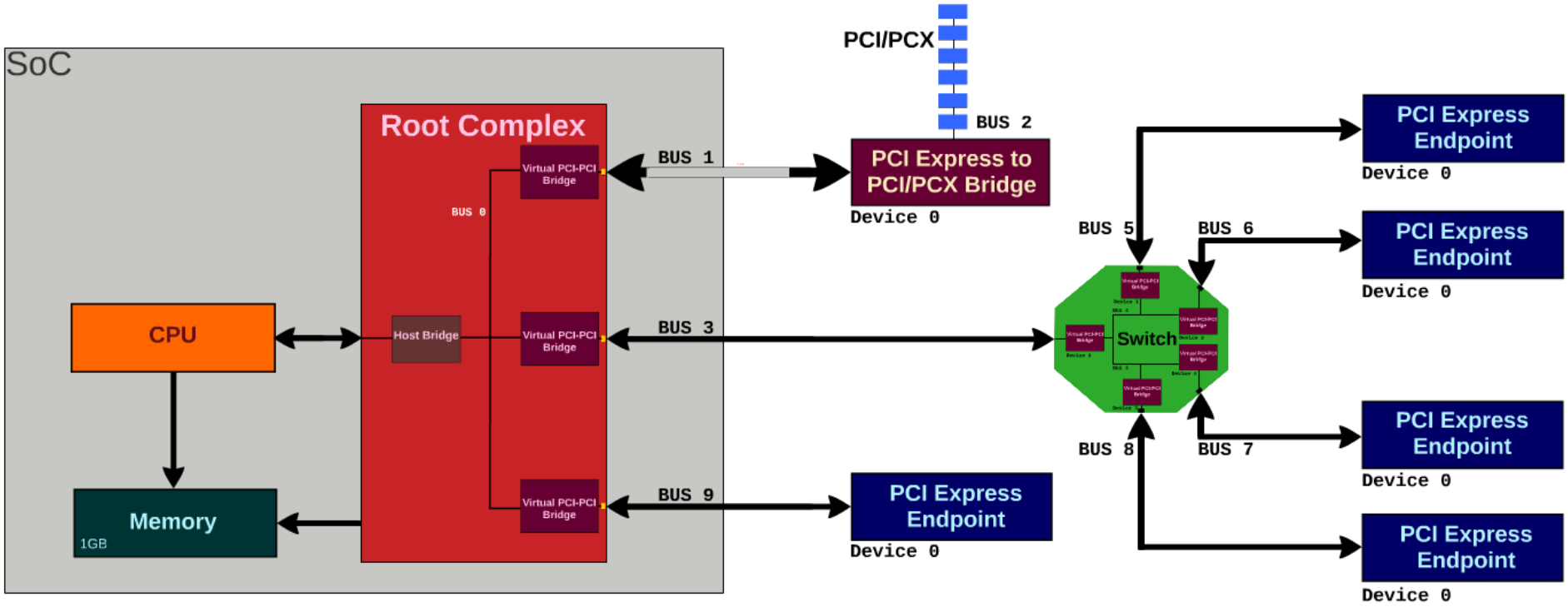
Topology


SoC
Root Complex
Virtual PCI-PCI Bridge
BUS 0
Host Bridge
Virtual PCI-PCI Bridge
Virtual PCI-PCI Bridge
CPU
Memory
1GB
BUS 1
BUS 3
BUS 9
PCI/PCX
BUS 2
PCI Express to PCI/PCX Bridge
Device 0
PCI Express Endpoint
Device 0
BUS 5
BUS 6
Switch
BUS 8
BUS 7
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0

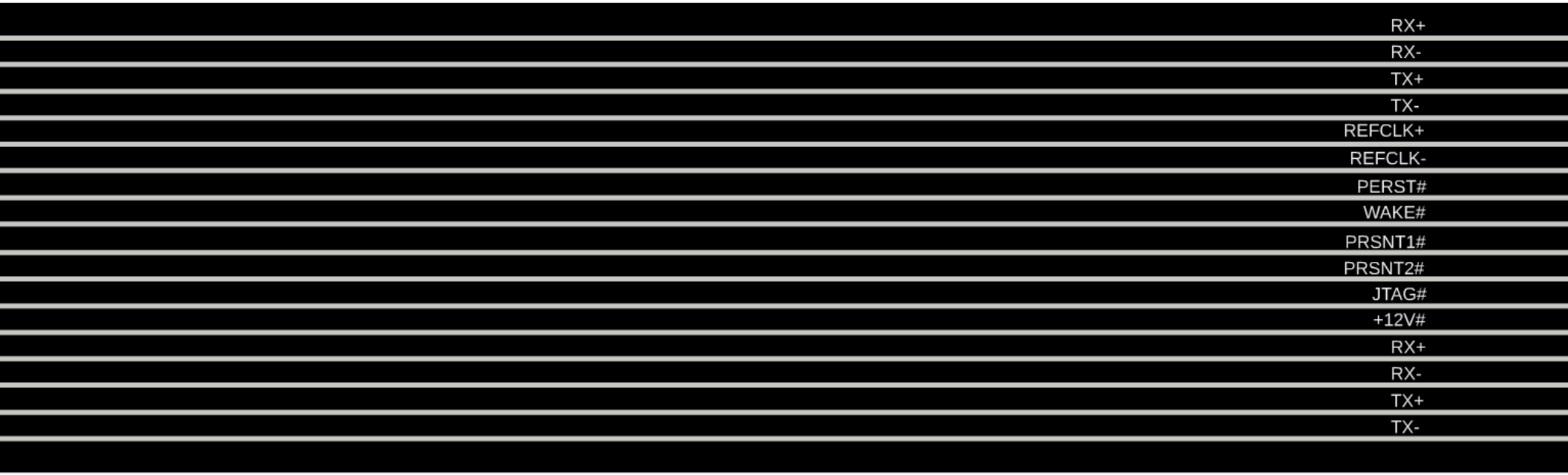

Root Complex
Virtual PCI-PCI Bridge
BUS 1
BUS 0
CPU
Host Bridge
Virtual PCI-PCI Bridge
BUS 3
emory
Virtual PCI-PCI Bridge
BUS 9


BUS 5
BUS 6
Virtual PCI-PCI Bridge
Device 1
BUS 4
Virtual PCI-PCI Bridge
Device 2
Virtual PCI-PCI Bridge
Switch
Virtual PCI-PCI Bridge
Device 0
Virtual PCI-PCI Bridge
Device 4
BUS 4
Virtual PCI-PCI Bridge
Device 3
BUS 8
BUS 7

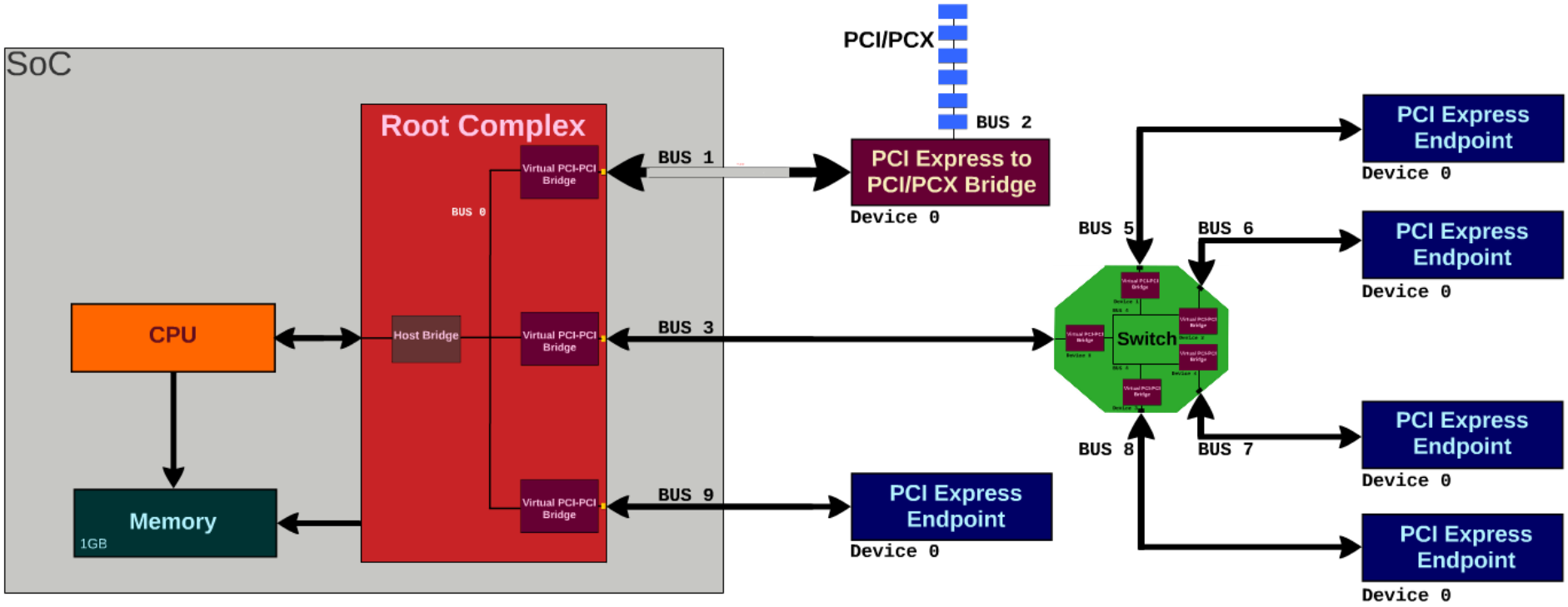
Topology


SoC
Root Complex
Virtual PCI-PCI Bridge
BUS 0
Host Bridge
Virtual PCI-PCI Bridge
Virtual PCI-PCI Bridge
CPU
Memory
1GB
BUS 1
BUS 3
BUS 9
PCI/PCX
BUS 2
PCI Express to PCI/PCX Bridge
Device 0
PCI Express Endpoint
Device 0
BUS 5
BUS 6
Switch
BUS 8
BUS 7
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0

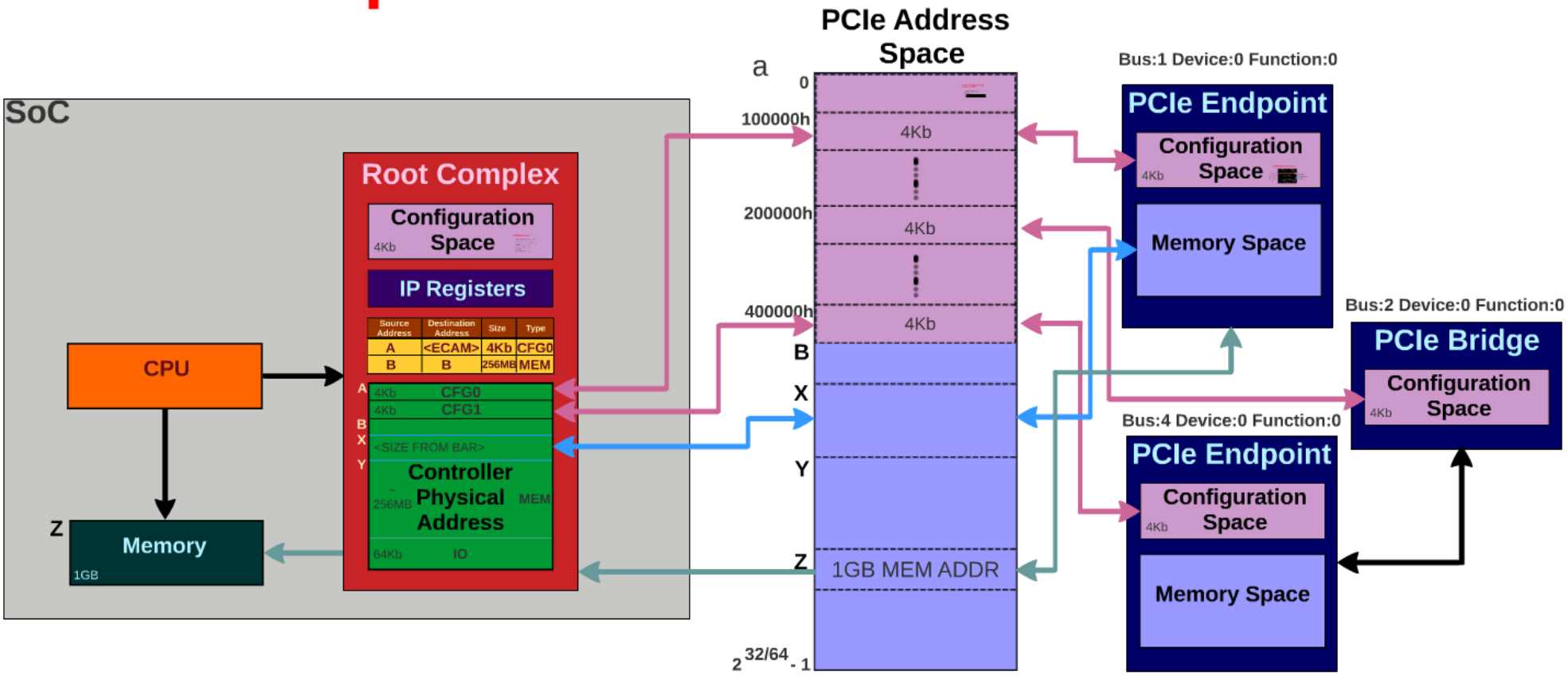
I/O Lines


RX+
RX-
TX+
TX-
REFCLK+
REFCLK-
PERST#
WAKE#
PRSNT1#
PRSNT2#
JTAG#
+12V#
RX+
RX-
TX+
TX-

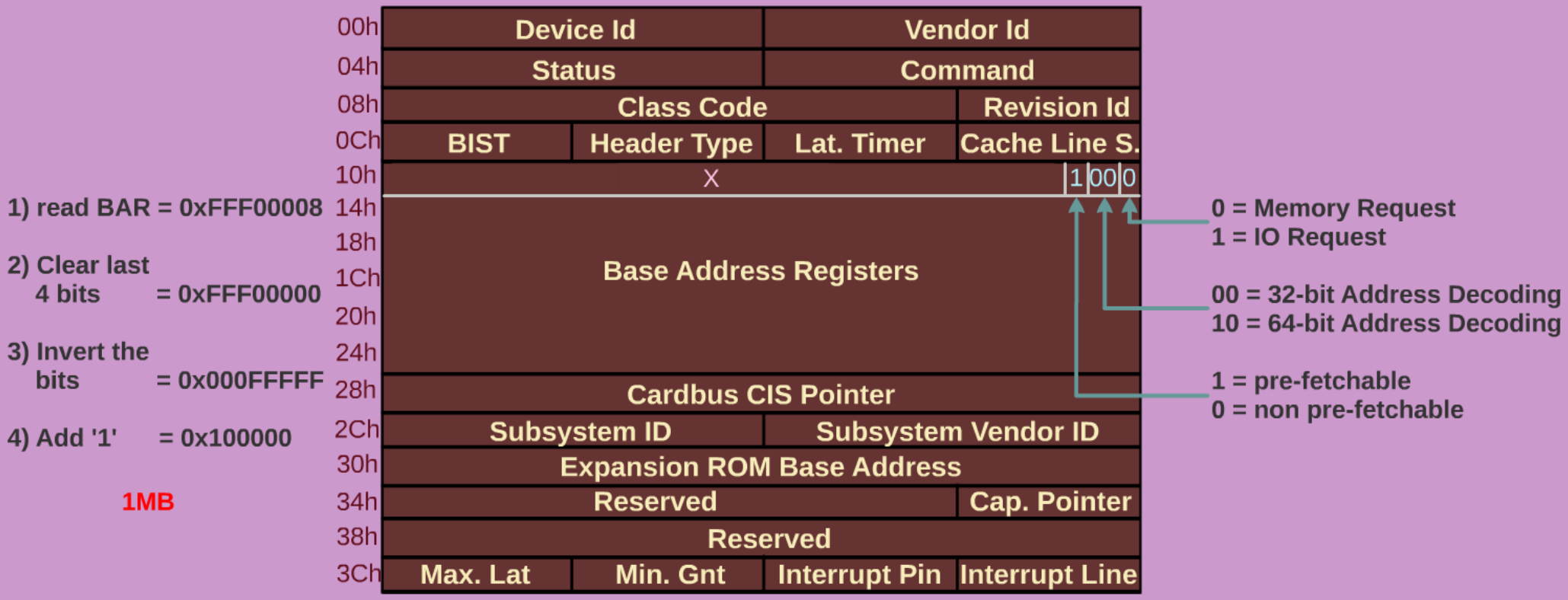
Topology


SoC
Root Complex
Virtual PCI-PCI Bridge
BUS 0
Host Bridge
Virtual PCI-PCI Bridge
Virtual PCI-PCI Bridge
CPU
Memory
1GB
BUS 1
BUS 3
BUS 9
PCI/PCX
BUS 2
PCI Express to PCI/PCX Bridge
Device 0
PCI Express Endpoint
Device 0
BUS 5
BUS 6
Switch
BUS 8
BUS 7
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0
PCI Express Endpoint
Device 0

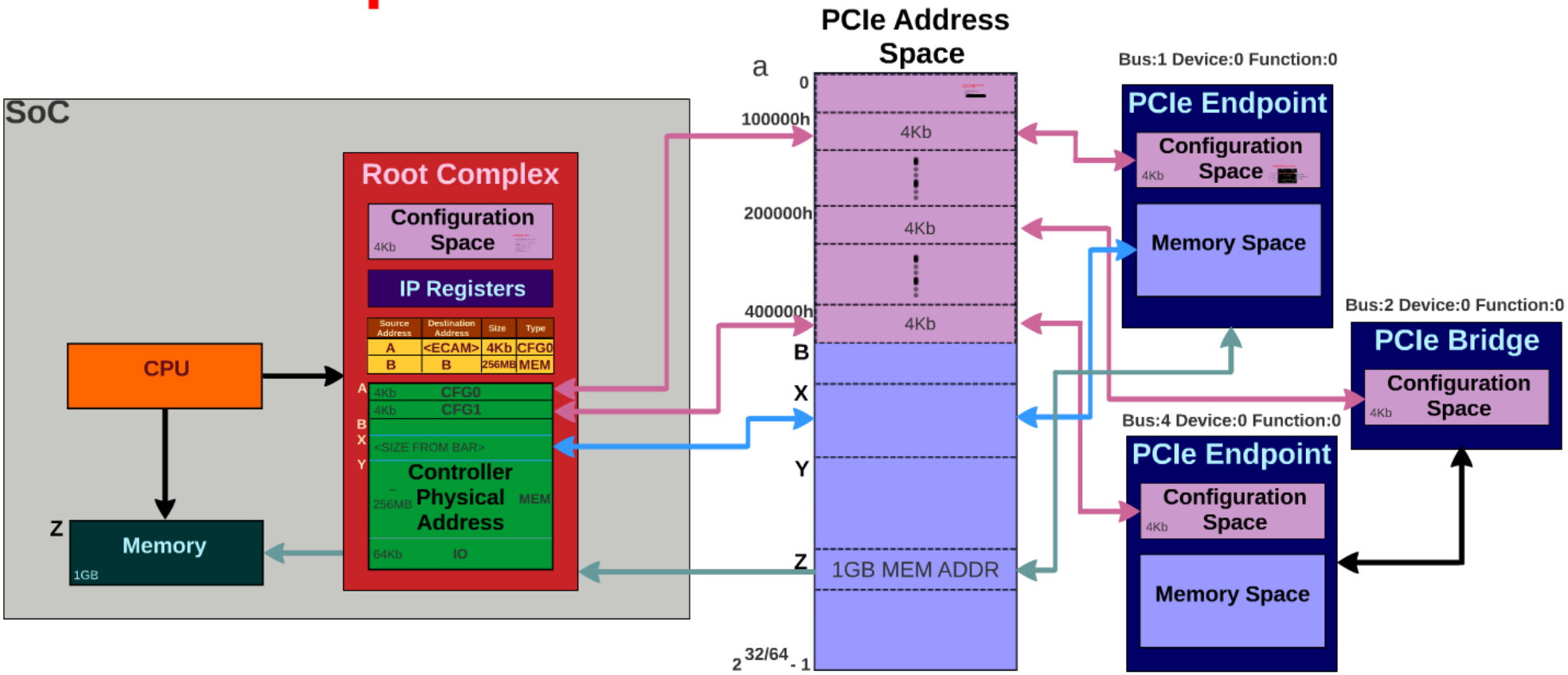
Address Space



Configuration Space Header

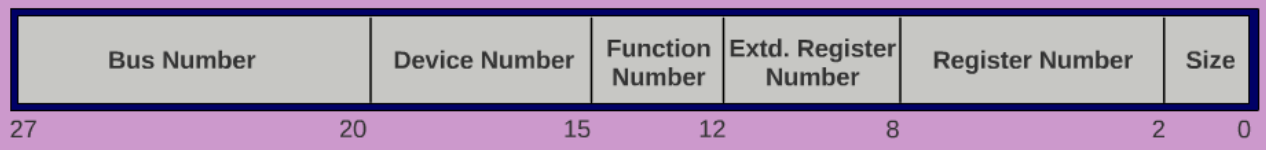


Address Space

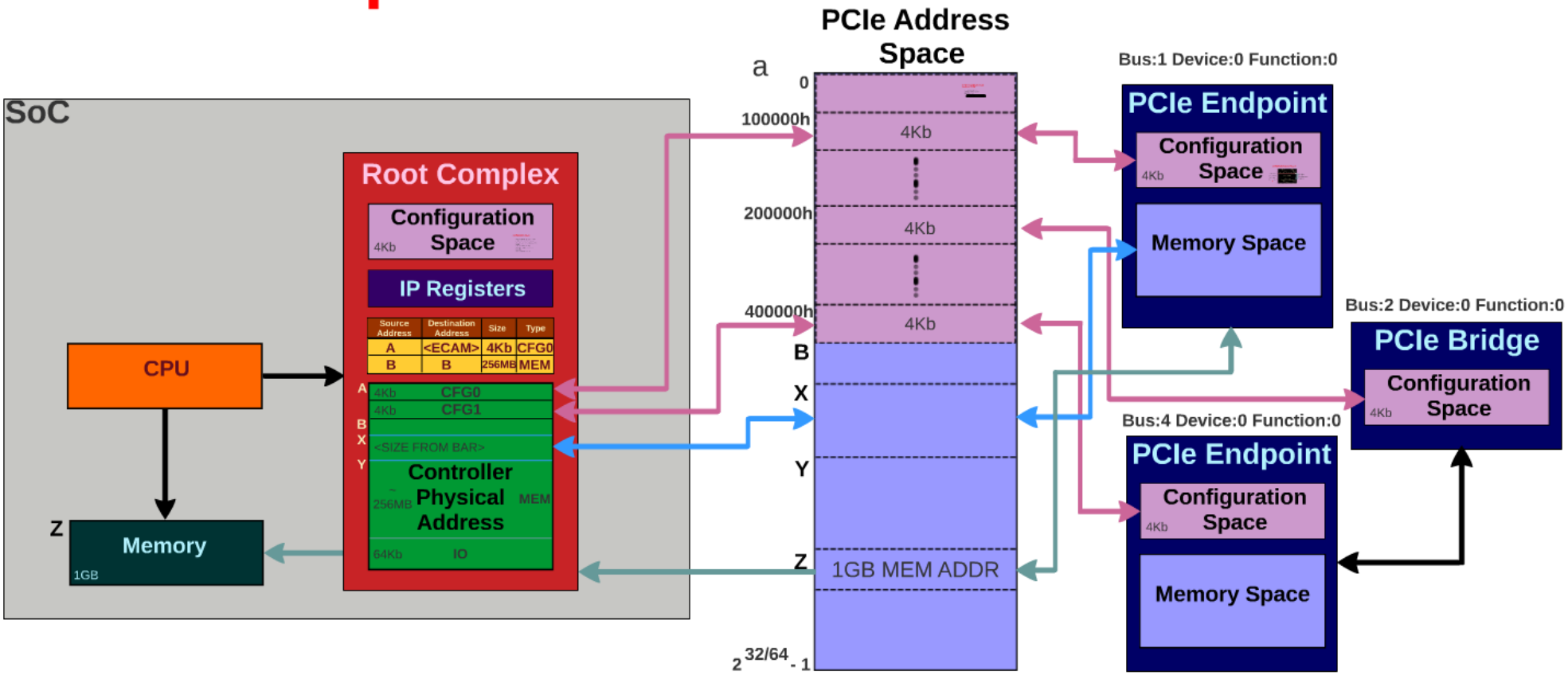


Enhanced Configuration Access Mechanism (ECAM)

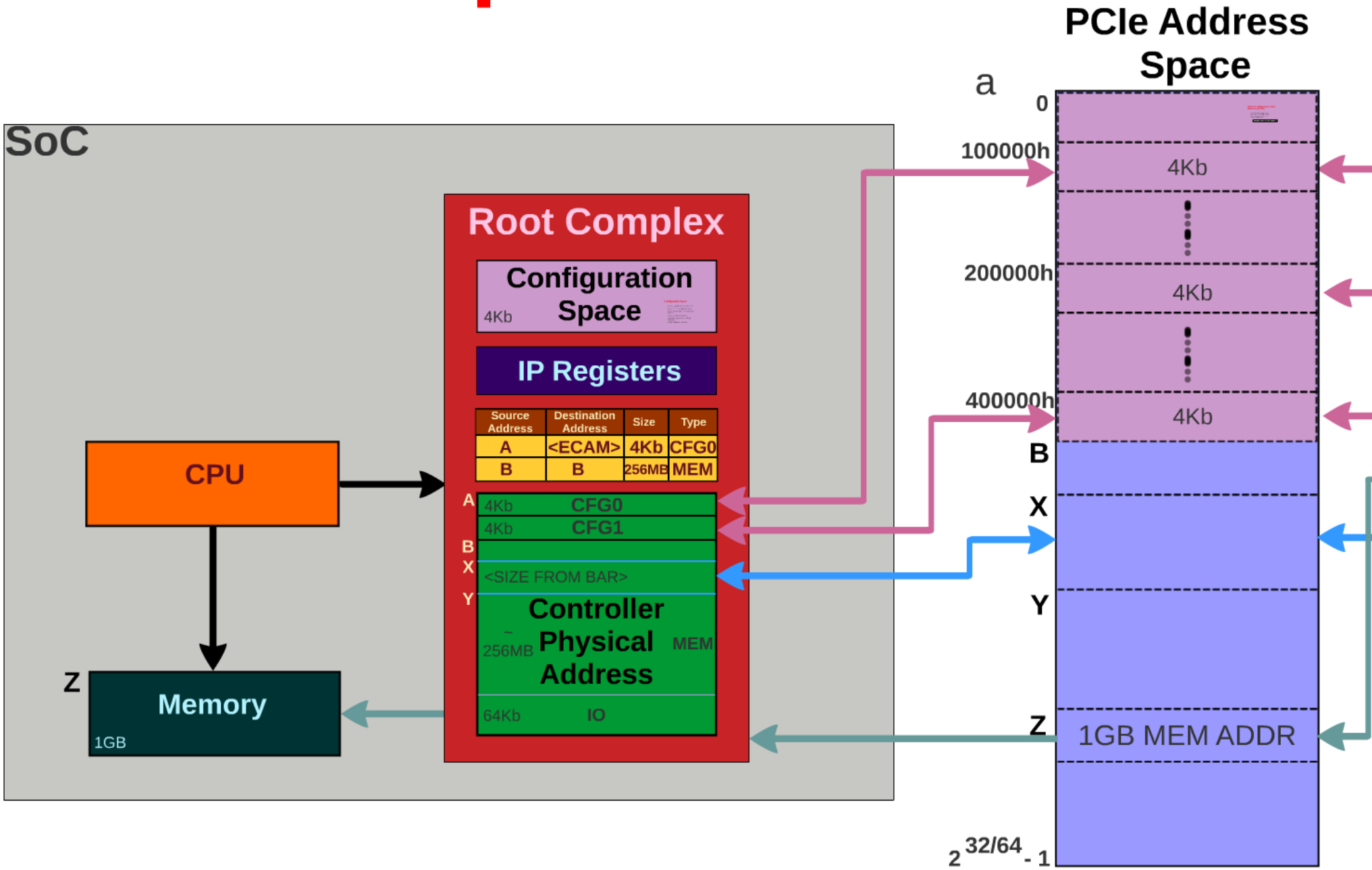
- Memory address (PCIe address space) determines configuration register accessed
- Function of bus number, device number, function and register number



Address Space



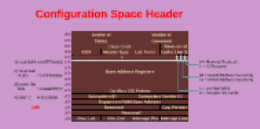
Address Space



PCIe Endpoint

Configuration Space

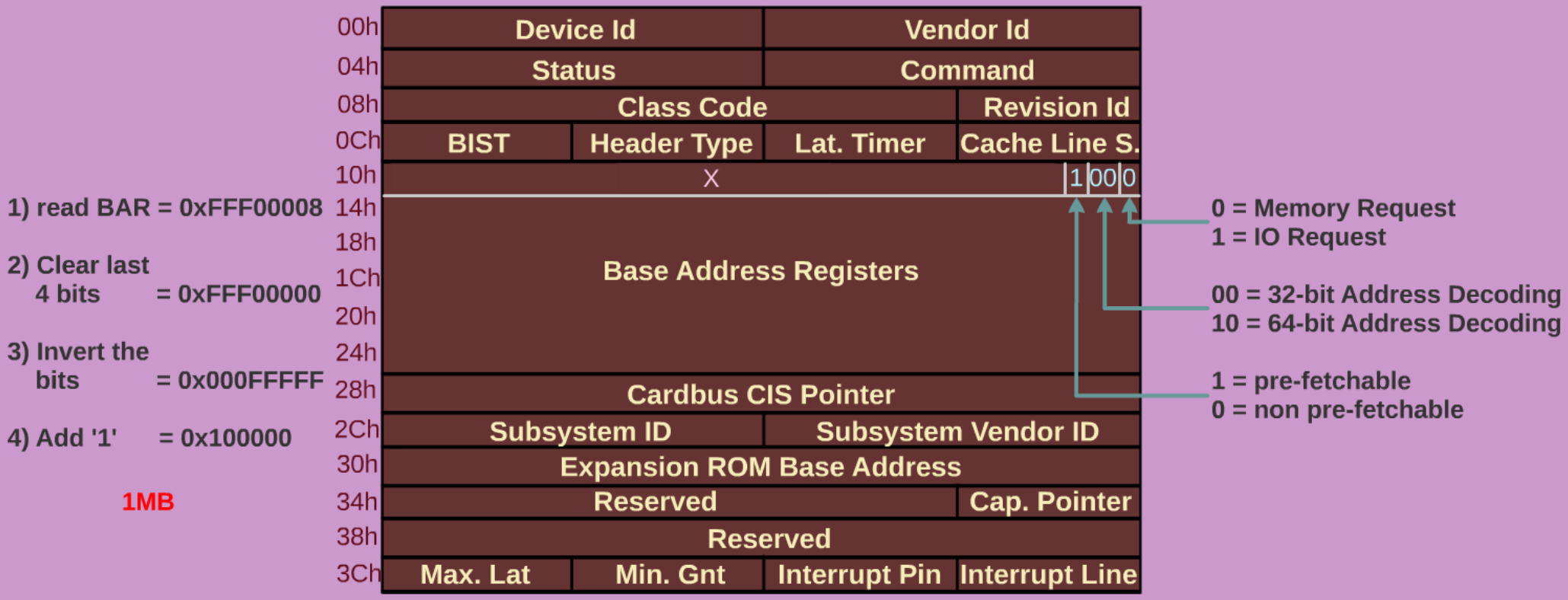
4Kb



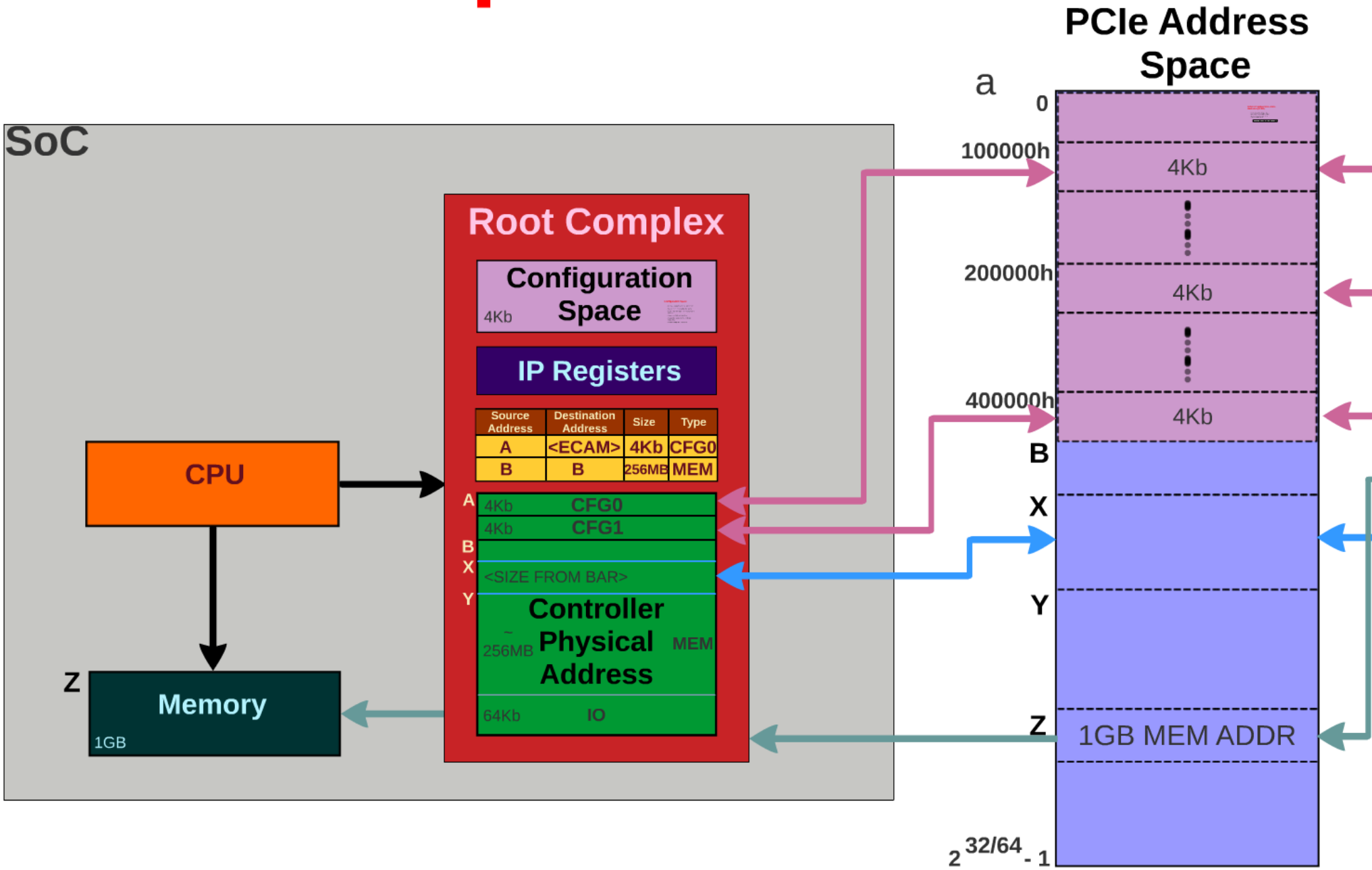
Memory Space

Bus:2

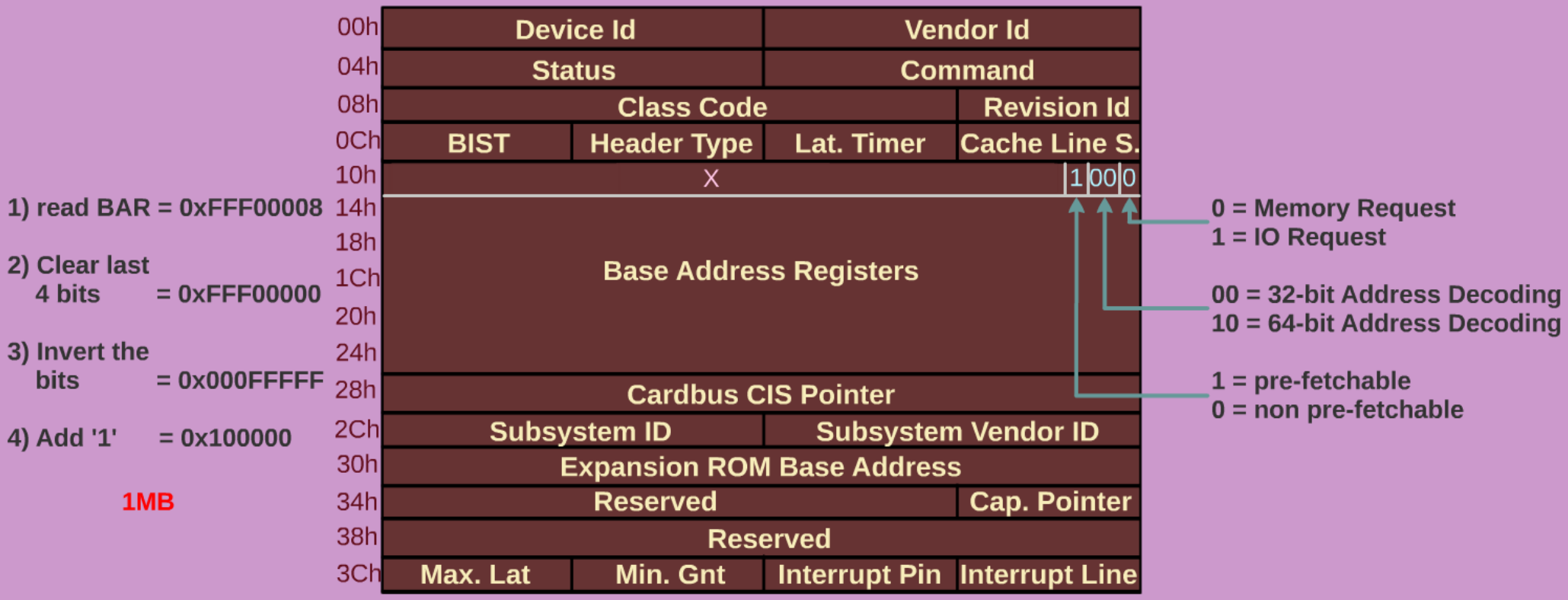
Configuration Space Header



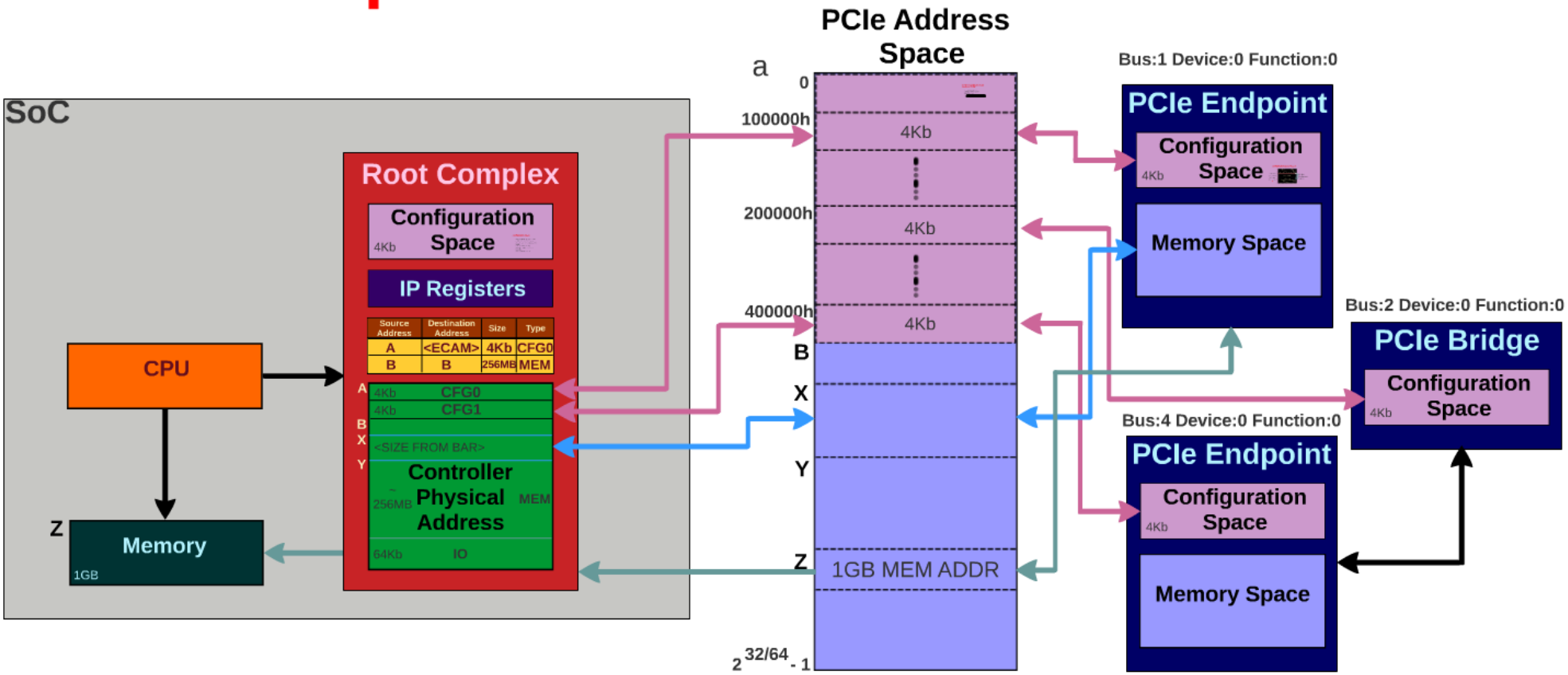
Address Space



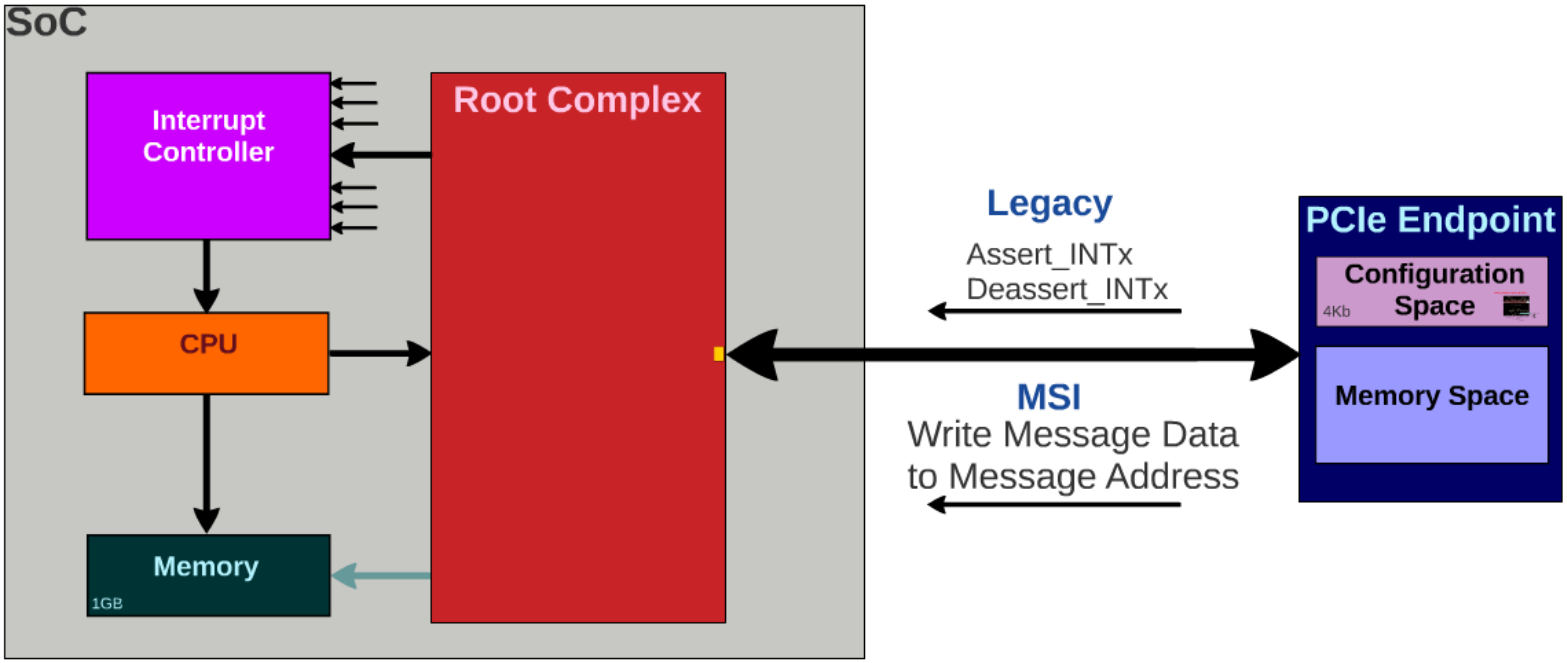
Configuration Space Header



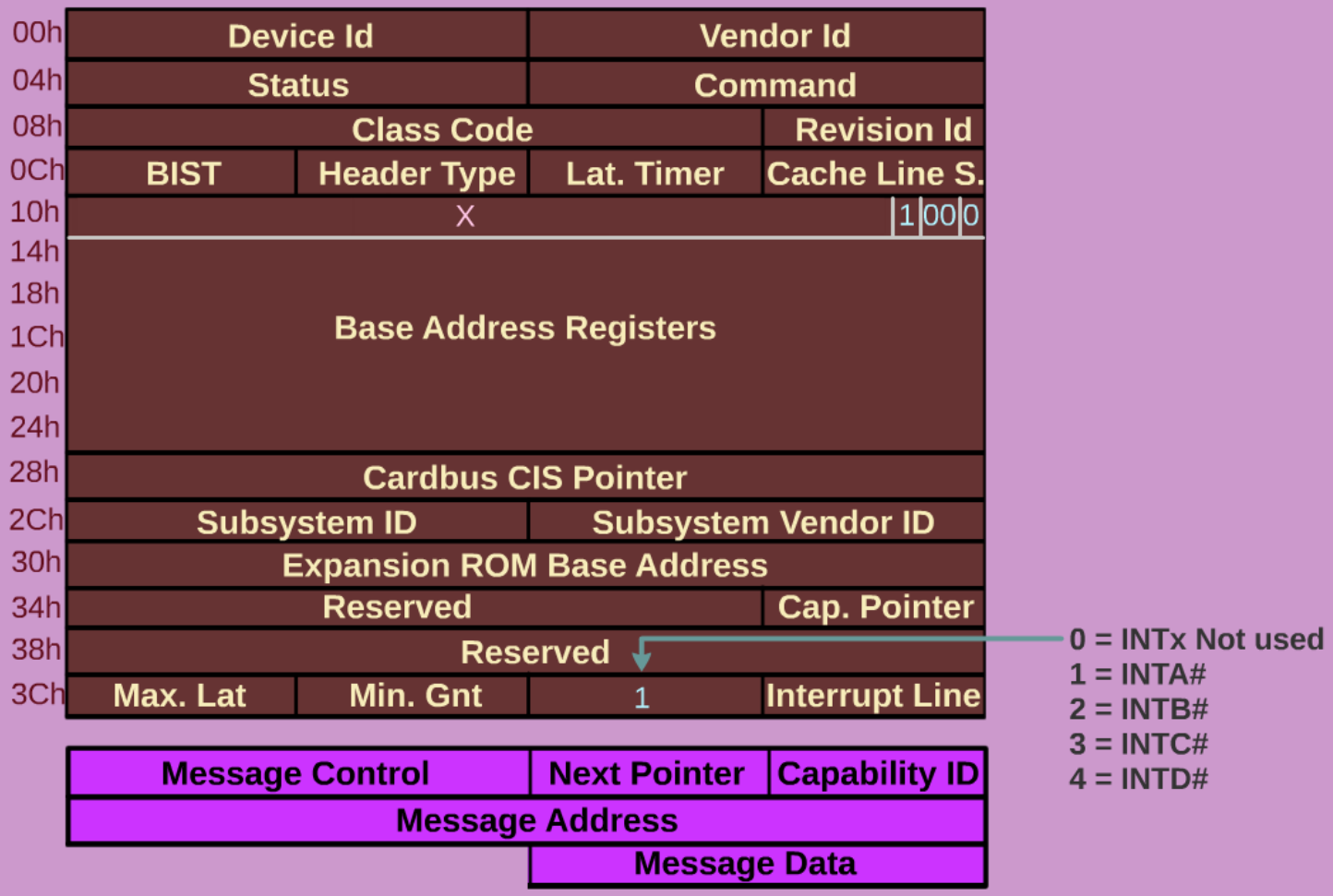
Address Space



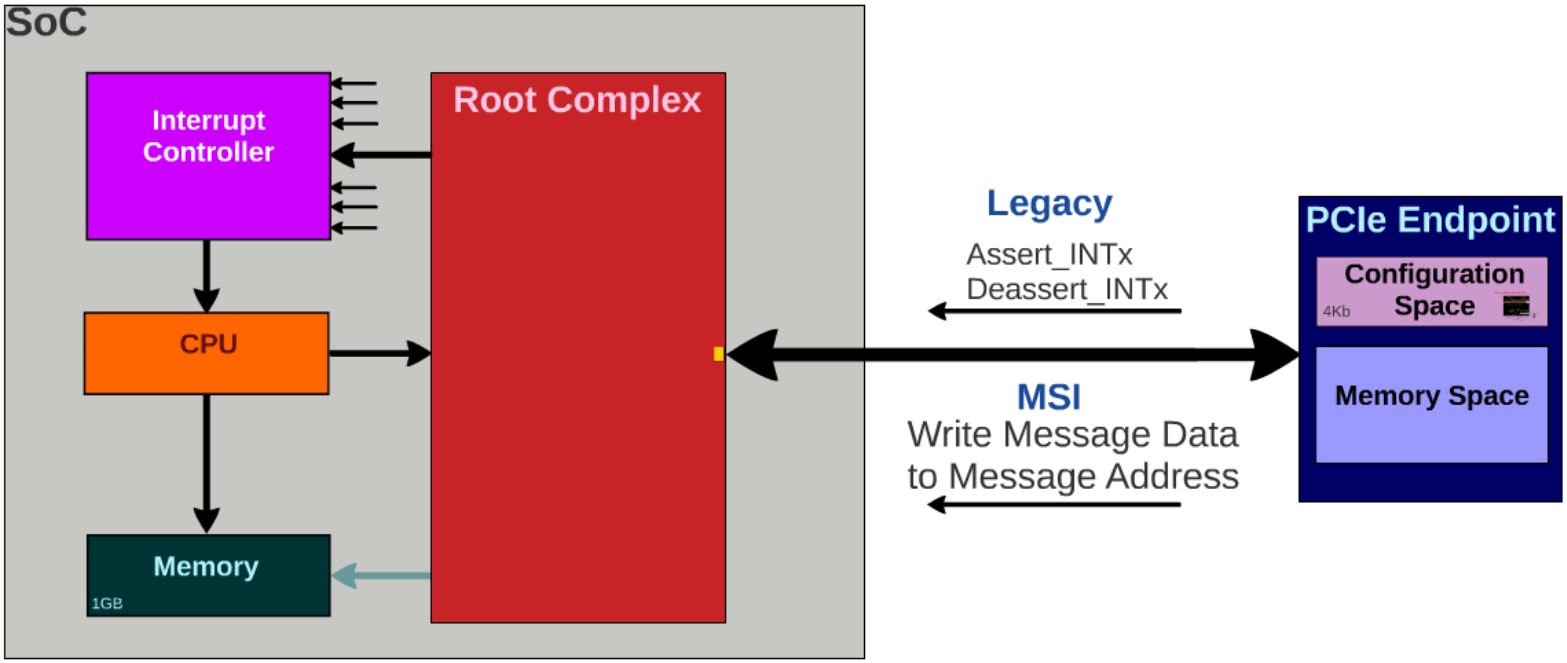
Interrupts


SoC
Interrupt Controller
CPU
Memory
1GB
Root Complex
Legacy
Assert_INTx
Deassert_INTx
MSI
Write Message Data
to Message Address
PCIe Endpoint
Configuration Space
4Kb
Memory Space

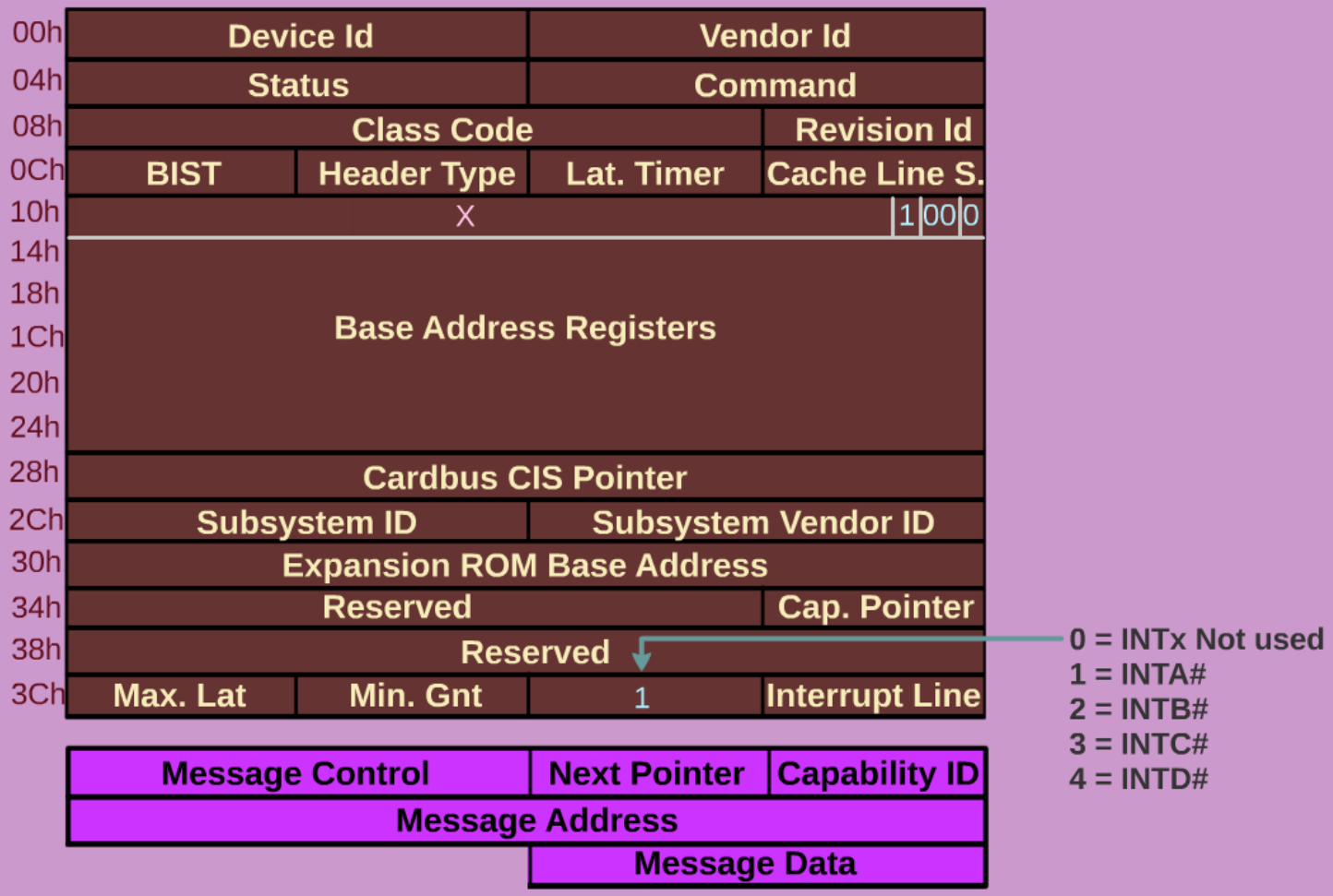
Configuration Space Header



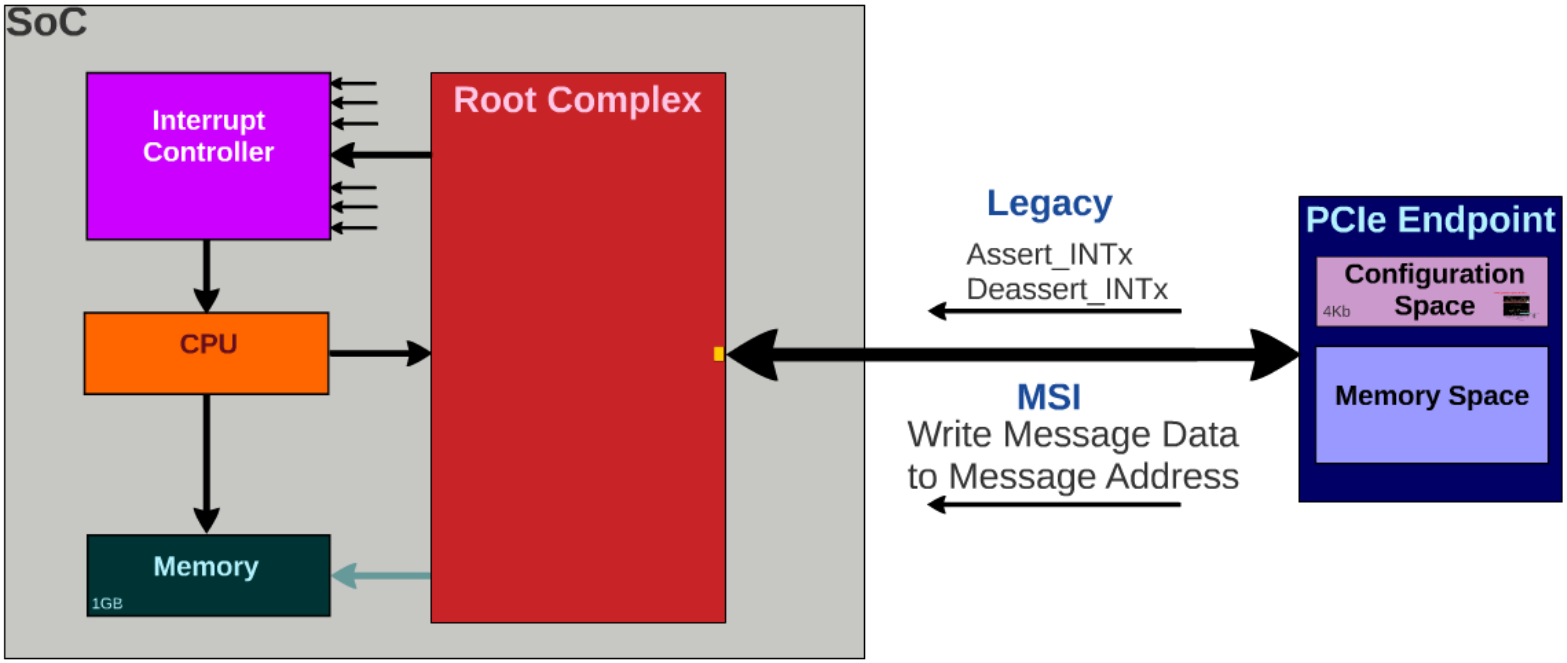
Interrupts


SoC
Interrupt Controller
CPU
Memory
1GB
Root Complex
Legacy
Assert_INTx
Deassert_INTx
MSI
Write Message Data
to Message Address
PCIe Endpoint
Configuration Space
4Kb
Memory Space

Configuration Space Header



Interrupts


SoC
Interrupt Controller
CPU
Memory
1GB
Root Complex
Legacy
Assert_INTx
Deassert_INTx
MSI
Write Message Data
to Message Address
PCIe Endpoint
Configuration Space
4Kb
Memory Space

Device Tree

```
pcie1: pcie@51000000 {
   compatible = "ti,dra7-pcie";
   reg = <0x51000000 0x2000>, <0x51002000 0x14c>, <A 0x2000>;
   reg-names = "rc_dbics", "ti_conf", "config";
   interrupts = <0 232 0x4>, <0 233 0x4>;
   device_type = "pci";
   #address-cells = <3>;
   #size-cells = <2>;
   ranges = <0x82000000 0 B B 0 0xfffd000>;
   interrupt-map-mask = <0 0 0 7>;
   interrupt-map = <0 0 0 1 &pcie1_intc 1>,
                   <0 0 0 2 &pcie1_intc 2>,
                   <0 0 0 3 &pcie1_intc 3>,
                   <0 0 0 4 &pcie1_intc 4>;
   pcie1_intc: interrupt-controller {
      interrupt-controller;
      #address-cells = <0>;
      #interrupt-cells = <1>;
   };
};
```

'ranges' Property

```
#address-cells = <3>;
#size-cells = <2>;
ranges = <0x82000000 0 B B 0 0xfffd000>;
```

PCI Address

Size

CPU Address

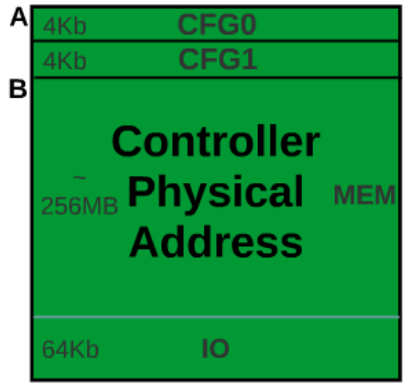
n	p	t	0 0 0	s s

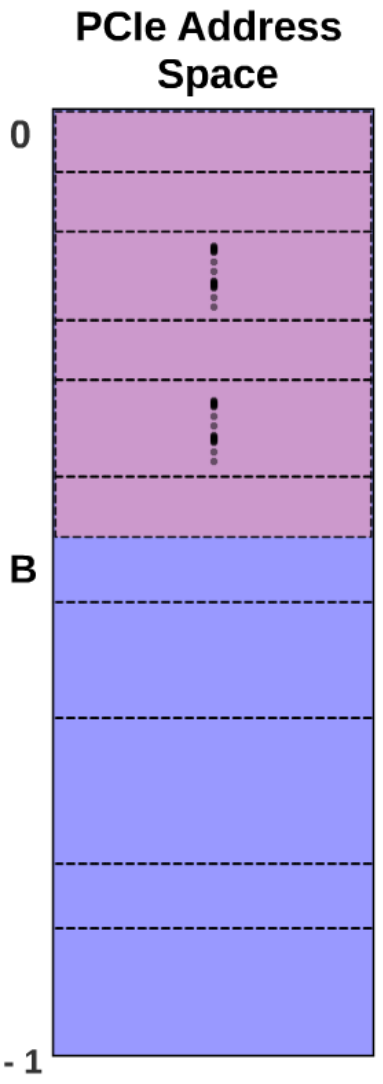
00: Configuration Space
01: I/O Space
10: 32 bit Memory Space
11: 64 bit Memory Space

prefetchable (cacheable)

relocatable

CPU	PCIE		
Source Address	Destination Address	Size	Type
B	B	~256MB	MEM

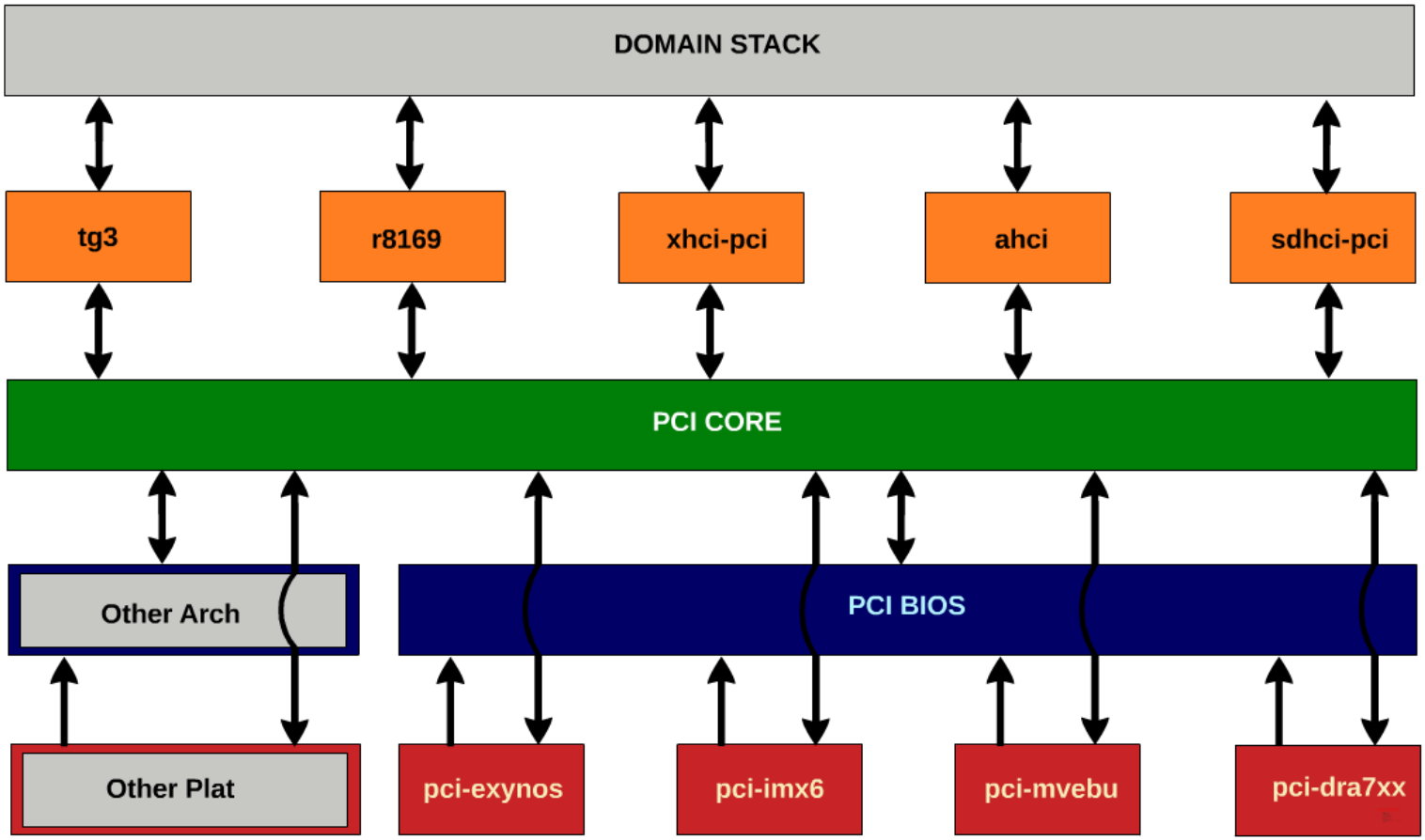




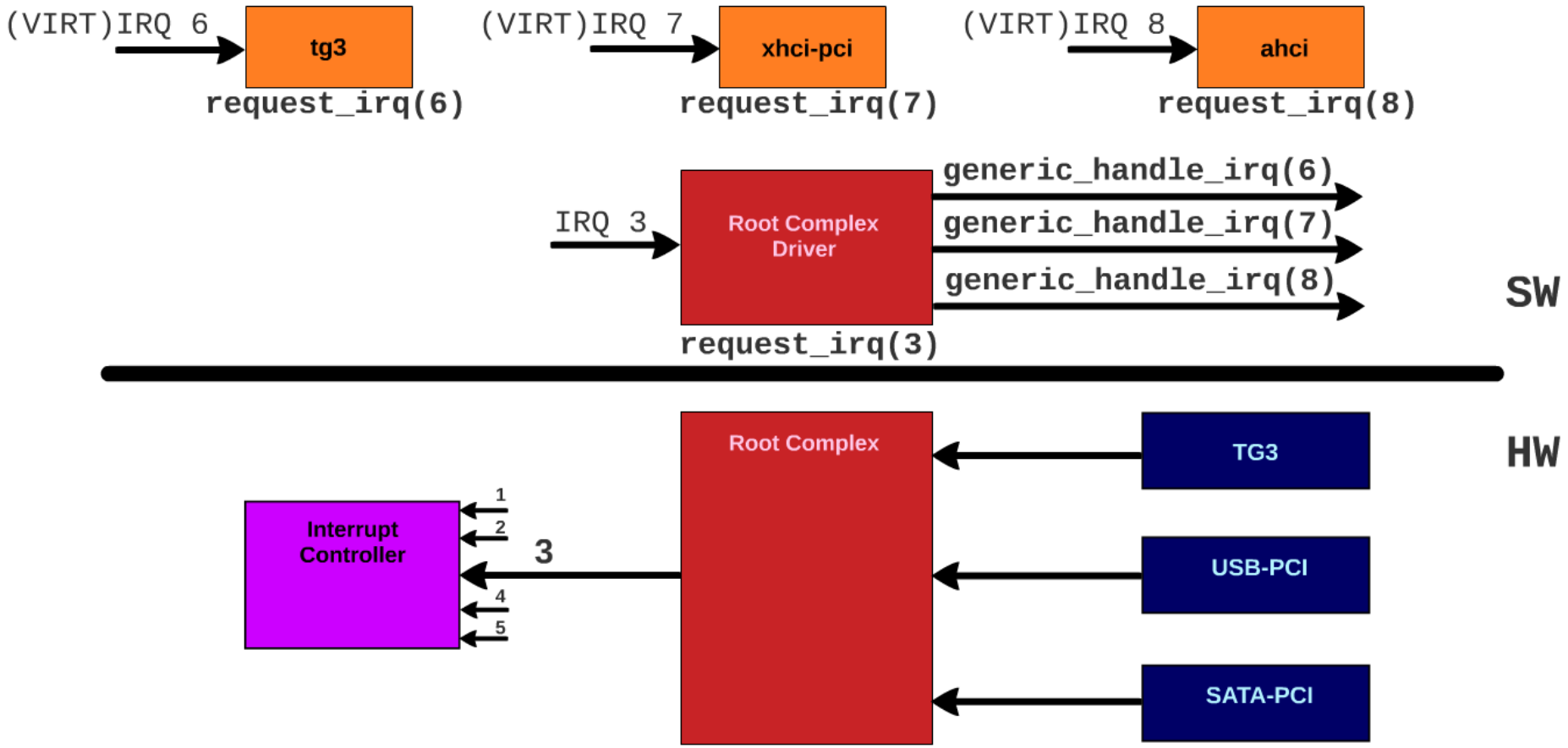
Device Tree

```
pcie1: pcie@51000000 {
   compatible = "ti,dra7-pcie";
   reg = <0x51000000 0x2000>, <0x51002000 0x14c>, <A 0x2000>;
   reg-names = "rc_dbics", "ti_conf", "config";
   interrupts = <0 232 0x4>, <0 233 0x4>;
   device_type = "pci";
   #address-cells = <3>;
   #size-cells = <2>;
   ranges = <0x82000000 0 B B 0 0xfffd000>;
   interrupt-map-mask = <0 0 0 7>;
   interrupt-map = <0 0 0 1 &pcie1_intc 1>,
                   <0 0 0 2 &pcie1_intc 2>,
                   <0 0 0 3 &pcie1_intc 3>,
                   <0 0 0 4 &pcie1_intc 4>;
   pcie1_intc: interrupt-controller {
      interrupt-controller;
      #address-cells = <0>;
      #interrupt-cells = <1>;
   };
};
```

Linux PCI(e) Subsystem



Interrupt Handling



lspci

- Displays all PCI buses, devices in a system

```
root@am57xx-evm:~# lspci
00:00.0 PCI bridge: Texas Instruments Device 8888 (rev 01)
01:00.0 PCI bridge: Pericom Semiconductor Device 2304 (rev 05)
02:01.0 PCI bridge: Pericom Semiconductor Device 2304 (rev 05)
02:02.0 PCI bridge: Pericom Semiconductor Device 2304 (rev 05)
03:00.0 SATA controller: ASMedia Technology Inc. ASM1062 Serial
ATA Controller (rev 01)
04:00.0 USB controller: Etron Technology, Inc. EJ168 USB 3.0 Host
Controller (rev 01)
```

- http://linuxcommand.org/man_pages/lspci8.html

Acknowledgements

- Jingoo Han, Pratyush Anand, Bjorn Helgaas
- Linux Community
- Texas Instruments
- Linux Foundation

References

- PCI Local Bus Specification 3.0
- PCI Express Base Specification 3.0
- PCI Express System Architecture (by Mindshare)

Happy Hacking!

Feedback:
kishon@ti.com
kishonvijayabraham@gmail.com

vigneshr@ti.com
vignesh.r.blr@gmail.com